Claims

1. A method of automatically and statically testing the design of a simulated integrated circuit (ASIC) includes the steps of:

simulating a circuit having a level containing a network of flip-flops;

putting the network in a reset state, in which each flip-flop would have an expected input and output state,

scanning the network and listing the input and output states;

listing the flip-flops with other than the expected input and output states as having potential faults;

examining each potential fault; and

modifying the circuit design so as to change the reset state of the flip-flop, or to add logic (A) to remedy the fault, so that when the modified design is retested by putting the network into a reset state and by releasing the reset state, each flip-flop will not go metastable when the reset state is released, a stable state being achieved in said flip-flop when its output value does not change on a subsequent active clock transition.

2. A method according to claim 1 wherein the flip-flops are of the D type.

3. A method according to claim 1 wherein the flip-flops are of the T type.

4. A method according to claim 1 wherein the flip-flops are of the J-K type.

5. A method according to any preceding claim wherein a PLI routine is linked with a simulator and a control program is then used to force a test netlist into a reset state;

a PLI code being called to log the state of the inputs and outputs of each flip-flop and the logged information being post-processed to create a list of flip-flops that have potential faults.

6. Apparatus for automatically and statically testing the design of a simulated integrated circuit, the apparatus including:

means for simulating an integrated circuit having a network of flip-flops;

means for scanning the network and for listing the input and output states;

means for listing the flip-flops which have potential faults;

means for examining each potential fault; and

means for modifying the circuit design to change the reset state of the flip-flop, or to add logic (A) to remedy the fault, so that when the modified design is retested by putting the network into a reset state and by releasing the reset state, each flip-flop will not go metastable when the reset state is released, a stable state being achieved in said flip-flop when its output value does not change on a subsequent active clock transition.

7. Apparatus according to claim 6 wherein a PLI routine is linked with a simulator and a control program is then used to force a test netlist into a reset state, a PLI code being called to log the state of the inputs and outputs of each flip-flop and the logged information being post-processed to create a list of flip-flops that have potential faults.